Editing Semantic Web Contents with Protégé: The OWL Plugin

Holger Knublauch
Stanford Medical Informatics
holger@smi.stanford.edu
OWL and the Semantic Web

• Web Ontology Language (OWL)
• Developed by a World Wide Web Consortium (W3C) working group
• Based on DAML+OIL

• Semantic Web Vision: To enable machines to **comprehend** semantic documents and data, not human speech and writings
• Habitat for Autonomous Agents
• OWL facilitates greater machine readability of Web content than XML
• Extends RDF and RDF Schema by providing additional vocabulary along with a formal semantics
OWL – Language Overview

• Classes
• Properties
  – DatatypeProperties
  – ObjectProperties
• Individuals

• Built-in ontology mapping support (equivalent classes, sameAs)
• Some other property types (e.g., symmetric, transitive, functional)

• Class Descriptions
  – can be used instead of named classes (e.g., to define superclasses)
  – classes are defined by the attributes of their members
    • enumerations red, green, or blue
    • restrictions all individuals that have at least 2 children
    • logical statements Person and not Student and not blue eyes
Description Logics (Example)

A BusyProfessor is a Professor with at least 3 students

A HappyPerson is a Person who has a workplace that is exciting and which lies in a region that is sunny

SunnyCities and DepressingCities are disjoint

Automated classification
OWL Plugin

- Extension of Protégé to allow editing OWL ontologies
- Project started April 2003, based on ideas from related work (OilTab, RDF and DAML+OIL backends)
- First alpha release ([http://protege.stanford.edu/plugins/owl](http://protege.stanford.edu/plugins/owl))

- Open and save OWL files in various formats (RDF/XML, N3, N-Triple)
- Graphical editors for class expressions
- Access to description logics inference components such as classifiers (not implemented yet)
- Integration of other OWL components based on Jena 2 API (HP Labs)
OWL Classes Tab

Protégé OWL Plugin

Holger Knublauch
OWL Expression Syntax

- RDF is the official OWL syntax:
  
  ```xml
  <owl:Class rdf:ID="Professor">
    <rdfs:subClassOf rdf:resource="#Person"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:about="#phD"/>
        </owl:onProperty>
        <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  ``

- OWL Abstract Syntax is an alternative but still rather verbose
### OWL Expression Syntax in Protégé

<table>
<thead>
<tr>
<th>OWL Element</th>
<th>Symbol</th>
<th>Key</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>allValuesFrom</td>
<td>∀</td>
<td>*</td>
<td>∀ children Male</td>
</tr>
<tr>
<td>someValuesFrom</td>
<td>∃</td>
<td>?</td>
<td>∃ children Lawyer</td>
</tr>
<tr>
<td>hasValue</td>
<td>$</td>
<td></td>
<td>rich true</td>
</tr>
<tr>
<td>cardinality</td>
<td>=</td>
<td>=</td>
<td>children = 3</td>
</tr>
<tr>
<td>minCardinality</td>
<td>≥</td>
<td>&gt;</td>
<td>children ≥ 3</td>
</tr>
<tr>
<td>maxCardinality</td>
<td>≤</td>
<td>&lt;</td>
<td>children ≤ 3</td>
</tr>
<tr>
<td>complementOf</td>
<td>¬</td>
<td>!</td>
<td>¬ Parent</td>
</tr>
<tr>
<td>intersectionOf</td>
<td>□</td>
<td>&amp;</td>
<td>Human □ Male</td>
</tr>
<tr>
<td>unionOf</td>
<td>♯</td>
<td></td>
<td></td>
</tr>
<tr>
<td>enumeration</td>
<td>{...}</td>
<td>{}</td>
<td>{male female}</td>
</tr>
</tbody>
</table>
OWL Expression Editor

Syntax checking

`MalePerson □ (rich ⊨ true)`

Error: Class or slot name expected at "Pe"

Word completion

`MalePerson □ (rich ⊨ true) □ Pe`

Potential completions: Person, PostDoc, Professor
Protégé OWL Plugin

**OWL and Protégé Comparison**

### Protégé
- Generic facet overriding
- :DEFAULTS
- :ROLE (abstract)
- :DOCUMENTATION overriding
- PAL Constraints
- Numeric min/max

### OWL
- Classes
- Transitive slots
- Slots/Properties
- Instances/Individuals
- InverseFunctional
- Template slot assignment/Domain
- Value types/Data types
- Restrictions = some facet overrides
- Metaclasses (instances of type "class")
- Equality and disjointness between classes, individuals
- Complex class expressions and anonymous classes can be used instead of named classes

Complex class expressions and anonymous classes can be used instead of named classes.
Protégé Metamodel (Version 2.0 alpha)
Protégé OWL Metamodel
Synchronizing Restrictions and Facet Overrides

DL Expression: "A Parent is a Person with at least one child."

<table>
<thead>
<tr>
<th>&lt;&lt; :OWL-NAMED-CLASS &gt;&gt;</th>
<th>&lt;&lt; :OWL-MINCARDI-RESTRICTION &gt;&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Person</strong></td>
<td></td>
</tr>
<tr>
<td>template slots: {children, ...}</td>
<td>template slots: {children}</td>
</tr>
<tr>
<td>facet overrides: {}</td>
<td>facet overrides: {children.minCardinality=1}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>&lt;&lt; :OWL-NAMED-CLASS &gt;&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Parent</strong></td>
</tr>
<tr>
<td>template slots: {...}</td>
</tr>
<tr>
<td>facet overrides: {children.minCardinality=1}</td>
</tr>
</tbody>
</table>
Protégé for OWL API

Protégé OWL Plugin

Holger Knublauch
Integrating OWL Components using the Jena API

Protégé with extended OWL Metamodel

Jena components (third-party)

Run-time mapping / Round-Trip Engineering

Protégé Model

Jena Model

ProtegeUpdater e.g. GraphEvent

JenaUpdater e.g. ClsEvent

OWL File
OWL – Jena Backend Architecture
Status and Current Work

• Mostly "just" an editor with load and save
• No OWL-specific reasoning implemented (no classifiers)
• Barely any "Save OWL file in format..."
• Neither ontology import nor different namespaces
• Not all OWL language elements can be edited with GUI
• Several TODO's and bugs (→ home page)

• No real hands-on experience (please try it and send feedback!)
• Based on other alpha software
• Expect a stable release in September

• Goals of early alpha release
  – People can experiment with OWL and our editor
  – Very few real OWL ontologies exist
  – Encourage third-party components (Jena API)
http://protege.stanford.edu/plugins/owl/

Many thanks to our partners in the CO-ODE Project at the University of Manchester (Alan Rector, Carole Goble, Sean Bechhofer, et al.)

Thanks to the Jena team at HP Labs in Bristol (Ian Dickinson & colleagues)
http://www.hpl.hp.com/semweb/

and (of course) to Ray Fergerson